Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claims 1.-9. (canceled).

Claim 10. (currently amended) A method performed by a custodian computing system, having memory, to share a secret S among n secret owners such that any k of the n secret owners may reconstruct the secret S, the method comprising the steps of:

choosing two large primes P and Q, such that PQ is greater than S when S is a number;

storing in the custodian computer system memory a product $N = PQ$;

storing a product $M = (P\text{-}1)(Q\text{-}1)$;

choosing n random numbers e_1 through e_n that are relatively prime to N;

choosing another random number e that is relatively prime to N;

choosing n numbers d_1 through d_n such that $e_i d_i \bmod M$ equals one for $1 \le i \le n$;

choosing another number d such that $ed \bmod M$ is equal to one;

storing a database of $\binom{n}{k}$ entries, wherein each entry is associated with a unique combination of the $\binom{n}{k}$ possible combinations of the k secret owners, and wherein a particular entry includes a value, c, that is the product of modulus M of d and the d_i values for i indices that correspond to the particular secret owners present in the unique combination for that particular entry, wherein c corresponds to modulus M of the product ~~kd_i~~ $\underline{kd_i}$;

computing S^e;

deleting from the custodian computer memory P, Q, and M;

distributing n secret owner pieces to each of the n secret owners, wherein each of the secret owner pieces includes S^e and one of the numbers e_1 through e_n;

deleting the secret S and e_1 through e_n, e, d_1 through d_n, and d;

receiving k secret owner values from a unique combination of k secret owners;

determining a the value c that is associated with the unique combination; and

determining the secret S using the value c retrieved from the database entry corresponding to the k secret owners whose secret owner pieces have been received and $S^e \bmod N$.

Claim 11. (previously presented) A method as in claim 10, wherein receiving k secret owner values from the unique combination of k secret owners comprises:

receiving a first of the n secret owner pieces from one of the n secret owners; and

computing and storing $S' = S^{ef} \bmod N$, where f represents the one of the numbers e_1 through e_n contained in the first of the n secret owner pieces.

Claim 12. (previously presented) A method as in claim 11, wherein receiving k secret owner values from the unique combination of k secret owners comprises:

receiving a second of the n secret owner pieces from another one of the n secret owners;

computing $S^q \bmod N$, where q represents the one of the numbers e_1 through e_n contained in the second of the n secret owner pieces; and replacing S' with $S^q \bmod N$.

Claim 13. (previously presented) A method as in claim 12, wherein receiving k secret owner values from the unique combination of k secret owners comprises:

each time another of the secret owner pieces is received from another one of the n secret owners;

computing $S^q \bmod N$, where q represents the one of the numbers e_1 through e_n contained in the another of the n secret owner pieces; and replacing S' with $S^q \bmod N$.

Claim 14. (previously presented) A method as in claim 13, further comprising the steps of:

after k secret owner pieces have been received,

retrieving from the database the value c from among the $\binom{n}{k}$ values, wherein the value c corresponds to the k secret owner pieces of the unique combination of k secret owners that were received by the custodian;

computing $S^e \bmod N$; and

replacing S' with $S^e \bmod N$.

Claim 15. (currently amended) A method performed by a custodian computing system, having memory, to share a secret S among n secret owners such that any k of the n secre owners may reconstruct the secret, the method comprising the steps of:

choosing two large primes P and Q, such that PQ is greater than S where S is a number;

storing in the custodian computer memory a product $N = PQ$;

storing a product $M = (P\text{-}1)(Q\text{-}1)$;

choosing n random numbers e_1 through e_n that are relatively prime to N;

choosing random numbers e and e' that are relatively prime to N;

choosing n numbers d_1 through d_n such that $e_i d_i \bmod M$ equals one for $1 \le i \le n$;

choosing numbers d and d' such that $ed \bmod M$ is equal to one and such that $e'd' \bmod M$ is equal to one;

storing a database of $\binom{n}{k}$ entries, wherein each entry is associated with a unique combination of the $\binom{n}{k}$ possible combinations of the k secret owners, and wherein a particular entry includes a value, c, that is the product of modulus M of d and the d_i values for i indices tha

correspond to the particular secret owners present in the unique combination for that particular entry, wherein c corresponds to modulus M of the product ~~kd1~~ kd_i;

computing $S^{ee'}$;

deleting from the custodian computer memory P, Q, and M;

distributing n secret owner pieces to each of the n secret owners, wherein each of the secret owner pieces includes $S^{ee'}$ and one of the numbers e_1 through e_n;

deleting the secret S and e_1 through e_n, e, d_1 through d_n, and d

receiving k secret owner values from a unique combination of k secret owners;

retrieving from the database the value c that is associated with the unique combination; and

determining the secret S using the value c and the k secret owner value.

Claim 16. (previously presented) A method as in claim 15, wherein receiving k secret owner values from the unique combination of k secret owners comprises:

receiving a first of the n secret owner pieces from one of the n secret owners; and

computing and storing $S' = S^{ee'f} \bmod N$, where f represents the one of the numbers e_1 through e_n contained in the first of the n secret owner pieces.

Claim 17. (previously presented) A method as in claim 16, wherein receiving k secret owner values from the unique combination of k secret owners comprises:

receiving a second of the n secret owner pieces from another one of the n secret owners;

computing $S^q \bmod N$, where q represents the one of the numbers e_1 through e_n contained in the second of the n secret owner pieces; and replacing S' with $S^q \bmod N$.

Claim 18. (previously presented) A method as in claim 17, wherein receiving k secret owner values from the unique combination of k secret owners comprises:

each time another of the secret owner pieces is received from another one of the n secret owners;

computing S^q mod N, where q represents the one of the numbers e_1 through e_n contained in the another of the n secret owner pieces; and replacing S' with S^q mod N.

Claim 19. (previously presented) A method as in claim 18, further comprising the steps of:

after k secret owner pieces have been received,

retrieving from the database the value c from among the $\binom{n}{k}$ values, wherein the value c corresponds to the k secret owner pieces from the unique combination of k secret owners that were received by the custodian;

computing S^c mod N;

replacing S' with S^c mod N;

computing $S^{d'}$ mod N; and

replacing S' with $S^{d'}$ mod N.

Claim 20. (currently amended) A method performed by a custodian computing system, having memory, to share a secret among n secret owners such that any k of the n secret owners may reconstruct the secret, the method comprising the steps of:

encrypting the secret so as to generate an encrypted secret;

deleting from the custodian computer memory the secret; and

performing a forward k out of n secret sharing algorithm on the encrypted secret so as to generate n secret owner pieces;

storing in a database a plurality of entries associated with a plurality of unique combinations of k secret owners of the n secret owners, wherein a particular entry includes a value, c, that is ~~the~~ a product of modulus M of d and ~~the~~ d_i values for i indices that correspond to the particular secret owners present in the unique combination for that particular entry, wherein c corresponds to modulus M of the product ~~kdi~~ kd_i;

distributing the n secret owner pieces to the n secret owners;

receiving k secret owner values from a unique combination of k secret owners;

retrieving from the database a value c that is associated with the unique combination;

performing a reverse *k* out of *n* secret sharing algorithm on the *k* secret owner pieces so as to recreate the encrypted secret using the value c; and

decrypting the encrypted secret so as to recreate the secret.

Claims 21. - 24. (canceled).

Claim 25. (original) A method as in claim 20, wherein the step of performing a forward *k* out of *n* secret sharing algorithm includes the steps of:

dividing the encrypted secret into *k* pieces; and

performing *n* polynomial evaluations at *n* points of a degree-*k* polynomial using the *k* pieces of the encrypted secret as polynomial coefficients;

wherein each of the *k* secret owner pieces includes a result of one of the *n* polynomial evaluations and a corresponding one of the *n* points.

Claim 26. (previously presented) A method as in claim 25, wherein the step of performing a reverse *k* out of *n* secret sharing algorithm includes the steps of generating a system of *k* linear equations and solving the system of *k* linear equations for the *k* pieces of the encrypted secret.

Claim 27. (previously presented) A method as in claim 26, further comprising the step of:

assembling the *k* pieces of the encrypted secret so as to recreate the encrypted secret.

Claims 28.-29. (canceled).

Claim 30. (currently amended) A computer readable storage medium having embodied thereon computer readable program code suitable for programming a computer to perform a method performed by a custodian to share a secret S among n secret owners such that any k of the n secret owners may reconstruct the secret, the method comprising the steps of:

choosing two large primes P and Q, such that PQ is greater than S where S is a number;

computing and storing a product $N = PQ$;

computing and storing a product $M = (P\text{-}1)(Q\text{-}1)$;

choosing n random numbers e_1 through e_n that are relatively prime to N;

choosing another random number e that is relatively prime to N;

choosing n numbers d_1 through d_n such that $e_i d_i \bmod M$ equals one for $1 \le i \le n$;

choosing another number d such that $ed \bmod M$ is equal to one;

storing a database of $\binom{n}{k}$ entries, wherein each entry is associated with a unique combination of the $\binom{n}{k}$ possible combinations of the k secret owners, and wherein a particular entry includes a value, c, that is the product of modulus M of d and the d_i values for i indices that correspond to the particular secret owners present in the unique combination for that particular entry, wherein c corresponds to modulus M of the product ~~kdi~~ kd_i;

computing S^e;

deleting P, Q, and M;

distributing n secret owner pieces to each of the n secret owners, wherein each of the secret owner pieces includes S^e and one of the numbers e_1 through e_n;

deleting the secret S and e_1 through e_n, e, d_1 through d_n, and d;

receiving k secret owner values from a unique combination of k secret owners;

retrieving from the database one of the values c that is associated with the unique combination; and

determining the secret S using the value c and the k secret owner values.

Claim 31. (currently amended) A computer readable storage medium having embodied thereon computer readable program code suitable for programming a computer to perform a method performed by a custodian to share a secret S among n secret owners such that any k of the n secret owners may reconstruct the secret, the method comprising the steps of:

choosing two large primes P and Q, such that PQ is greater than S where S is a number;

storing a product $N = PQ$;

storing a product $M = (P\text{-}1)(Q\text{-}1)$;

choosing n random numbers e_1 through e_n that are relatively prime to N;

choosing random numbers e and e' that are relatively prime to N;

choosing n numbers d_1 through d_n such that $e_i d_i \bmod M$ equals one for $1 \le i \le n$;

choosing numbers d and d' such that $ed \bmod M$ is equal to one and such that $e'd' \bmod M$ is equal to one;

generating and storing a database of $\binom{n}{k}$ values, where each value is the product of d and a unique k of the d_i numbers for $1 \le i \le n$, wherein each value is associated with a unique combination of k secret owners of the n secret owners;

storing a database of $\binom{n}{k}$ entries, wherein each entry is associated with a unique combination of the $\binom{n}{k}$ possible combinations of the k secret owners, and wherein a particular entry includes a value, c, that is the product of modulus M of d and the d_i values for i indices that correspond to the particular secret owners present in the unique combination for that particular entry, wherein c corresponds to modulus M of the product ~~kd_i~~ $\underline{kd_i}$;

computing $S^{ee'}$;

deleting P, Q, and M;

distributing n secret owner pieces to each of the n secret owners, wherein each of the secret owner pieces includes $S^{ee'}$ and one of the numbers e_1 through e_n;

deleting the secret S and e_1 through e_n, e, d_1 through d_n, and d;

receiving k secret owner values from a unique combination of k secret owners;

retrieving from the database one of the values c that is associated with the unique combination; and

determining the secret S using the value c and the k secret owner values.

Claim 32. (previously presented) A computer readable storage medium having embodied thereon computer readable program code suitable for programming a computer to perform a method performed by a custodian to share a secret among n secret owners such that any k of the n secret owners may reconstruct the secret, the method comprising the steps of:

encrypting the secret so as to generate an encrypted secret;

deleting the secret;

performing a forward k out of n secret sharing algorithm on the encrypted secret so as to generate n secret owner pieces;

storing in a database a plurality of entries associated with a plurality of unique combinations of k secret owners of the n secret owners, wherein a particular entry includes a value, c, that is ~~the~~ a product of modulus M of d and ~~the~~ d_i values for i indices that correspond to the particular secret owners present in the unique combination for that particular entry, wherein i corresponds to modulus M of the product ~~kdi~~ kd_i;

distributing the n secret owner pieces to the n secret owners;

receiving k secret owner values from a unique combination of k secret owners;

retrieving from the database one of the values c that is associated with the unique combination;

performing a reverse k out of n secret sharing algorithm on the k secret owner pieces so as to recreate the encrypted secret using the value c; and

decrypting the encrypted secret so as to recreate the secret.

Claims 33.-34. (canceled).

Claim 35. (currently amended) A computer comprising a processor and a computer readable storage medium coupled to the processor having embodied thereon processo readable program code suitable for programming a computer to perform a method performed by a custodian to share a secret S among n secret owners such that any k of the n secret owners may reconstruct the secret, the method comprising the steps of:

choosing two large primes P and Q, such that PQ is greater than S where S is a number;

storing a product $N = PQ$;

storing a product $M = (P\text{-}1)(Q\text{-}1)$;

choosing n random numbers e_1 through e_n that are relatively prime to N;

choosing another random number e that is relatively prime to N;

choosing n numbers d_1 through d_n such that $e_i d_i \bmod M$ equals one for $1 \le i \le n$;

choosing another number d such that $ed \bmod M$ is equal to one;

storing a database of $\binom{n}{k}$ entries, wherein each entry is associated with a unique combination of the $\binom{n}{k}$ possible combinations of the k secret owners, and wherein a particular entry includes a value, c, that is the product of modulus M of d and the d_i values for i indices that correspond to the particular secret owners present in the unique combination for that particular entry, wherein c corresponds to modulus M of the product ~~kdi~~ kd_i;

computing S^e;

deleting P, Q, and M;

distributing n secret owner pieces to each of the n secret owners, wherein each of the secret owner pieces includes S^e and one of the numbers e_1 through e_n;

deleting the secret S and e_1 through e_n, e, d_1 through d_n, and d;

receiving k secret owner values from a unique combination of k secret owners;

retrieving from the database one of the a values c that is associated with the unique combination; and

determining the secret S using the value c and the k secret owner values.

Claim 36. (currently amended) A computer comprising a processor and a computer readable storage medium coupled to the processor having embodied thereon processor readable program code suitable for programming the computer to perform a method performed by a custodian to share a secret S among n secret owners such that any k of the n secret owners may reconstruct the secret, the method comprising the steps of:

choosing two large primes P and Q, such that PQ is greater than S where S is a number;

storing a product $N = PQ$;

storing a product $M = (P\text{-}1)(Q\text{-}1)$;

choosing n random numbers e_1 through e_n that are relatively prime to N;

choosing random numbers e and e' that are relatively prime to N;

choosing n numbers d_1 through d_n such that $e_i d_i \bmod M$ equals one for $1 \le i \le n$;

choosing numbers d and d' such that $ed \bmod M$ is equal to one and such that $e'd' \bmod M$ is equal to one;

generating and storing a database of $\binom{n}{k}$ values, where each value is the product of d and a unique k of the d_i numbers for $1 \le i \le n$, wherein each value is associated with a unique combination of k secret owners of the n secret owners;

storing a database of $\binom{n}{k}$ entries, wherein each entry is associated with a unique combination of the $\binom{n}{k}$ possible combinations of the k secret owners, and wherein a particular entry includes a value, c, that is the product of modulus M of d and the d_i values for i indices that correspond to the particular secret owners present in the unique combination for that particular entry, wherein c corresponds to modulus M of the product ~~kdi~~ kd_i;

computing $S^{ee'}$;

deleting P, Q, and M;

distributing n secret owner pieces to each of the n secret owners, wherein each o the secret owner pieces includes $S^{ee'}$ and one of the numbers e_1 through e_n;

deleting the secret S and e_1 through e_n, e, d_1 through d_n, and d;

receiving k secret owner values from a unique combination of k secret owners;

retrieving from the database one of the values c that is associated with the unique combination; and

determining the secret S using the value c and the k secret owner values.

Claim 37. (currently amended) A computer comprising a processor and a computer readable storage medium coupled to the processor having embodied thereon processor readable program code suitable for programming the computer to perform a method performed by a custodian to share a secret among n secret owner such that any k of the n secret owners may reconstruct the secret, the method comprising the steps of:

encrypting the secret so as to generate an encrypted secret;

deleting the secret;

performing a forward k out of n secret sharing algorithm on the encrypted secret so as to generate n secret owner pieces;

storing in a database a plurality of entries associated with a plurality of unique combinations of k secret owners of the n secret owners, wherein a particular entry includes a value, c, that is ~~the~~ a product of modulus M of d and ~~the~~ d_i values for i indices that correspond to the particular secret owners present in the unique combination for that particular entry, wherein corresponds to modulus M of the product ~~kdi~~ kd_i;

distributing the n secret owner pieces to the n secret owners;

receiving k secret owner values from a unique combination of k secret owners;

retrieving from the database one of the values c that is associated with the unique combination;

performing a reverse k out of n secret sharing algorithm on the k secret owner pieces so as to recreate the encrypted secret using the value c; and

decrypting the encrypted secret so as to recreate the secret.